EdSIDH: Supersingular Isogeny Diffie-Hellman Key Exchange on Edwards Curves

Reza Azarderakhsh[1], Elena Bakos Lang[2], David Jao[2,3,4], and Brian Koziel[5]

[1] Department of Computer and Electrical Engineering and Computer Science, Florida Atlantic University, Boca Raton, Florida
[2] Department of Combinatorics and Optimization, University of Waterloo, Waterloo, Ontario, N2L 3G1, Canada
[3] Centre for Applied Cryptographic Research, University of Waterloo, Waterloo, Ontario, N2L 3G1, Canada
[4] evolutionQ, Inc., Waterloo, Ontario, Canada
[5] Texas Instruments, Dallas, Texas

Abstract. Problems relating to the computation of isogenies between elliptic curves defined over finite fields have been studied for a long time. Isogenies on supersingular elliptic curves are a candidate for quantum-safe key exchange protocols because the best known classical and quantum algorithms for solving well-formed instances of the isogeny problem are exponential. We propose an implementation of supersingular isogeny Diffie-Hellman (SIDH) key exchange for complete Edwards curves. Our work is motivated by the use of Edwards curves to speed up many cryptographic protocols and improve security. Our work does not actually provide a faster implementation of SIDH, but the use of complete Edwards curves and their complete addition formulae provides security benefits against side-channel attacks. We provide run time complexity analysis and operation counts for the proposed key exchange based on Edwards curves along with comparisons to the Montgomery form.

Keywords: Edwards curves, isogeny arithmetic, supersingular isogeny Diffie-Hellman key exchange

1 Introduction

According to our current understanding of the laws of quantum mechanics, quantum computers based on quantum phenomena offer the possibility of solving certain problems much more quickly than is possible on any classical computer. Included among these problems are almost all of the mathematical problems upon which currently deployed public-key cryptosystems are based. NIST has recently announced plans for transitioning to post-quantum cryptographic protocols, and organized a standardization process for developing such cryptosystems [9]. One of the candidates in this process is Jao and De Feo's Supersingular Isogeny Diffie-Hellman (SIDH) proposal [14], which is based on the path-finding problem in isogeny graphs of supersingular elliptic curves [8, 10]. Isogenies are

a special kind of morphism of algebraic curves, which have been studied extensively in pure mathematics but only recently proposed for use in cryptography. We believe isogeny-based cryptosystems offer several advantages compared to other approaches for post-quantum cryptography:

- Their security level is determined by a simple choice of a single public parameter. The temptation in cryptography is always to cut parameter sizes down to the bare minimum security level, for performance reasons. By reducing the number of security-sensitive parameters down to one, it becomes impossible to accidentally choose one parameter too small in relation to the others (which harms security), or too large (which harms performance).
- They achieve the smallest public key size among those post-quantum cryptosystems which were proposed to NIST [30, Table 5.9].
- They are based on number-theoretic complexity assumptions, for which there is already a large base of existing research, activity, and community expertise.
- Implementations can leverage existing widely deployed software libraries to achieve necessary features such as side-channel resilience.

Relative to other post-quantum candidates, the main practical limitation of SIDH currently lies in its performance which requires more attention from cryptographic engineers.

The majority of speed-optimized SIDH implementations (in both hardware and software platforms) use Montgomery curves [12, 16, 11, 14, 13, 2, 1, 22, 17, 23, 26, 25, 32, 24], which are a popular choice for cryptographic applications due to their fast curve and isogeny arithmetic. Only [27] is an exception as it considers a hybrid Edwards-Montgomery SIDH scheme that still uses isogenies over Montgomery curves. Alternative models for elliptic curves have been studied for fast computation such as Edwards curves, whose complete addition law presents security and speed benefits for the implementation of various cryptographic protocols. Edwards curves and Montgomery curves share many characteristics, as there is a birational equivalence between the two families of curves. Edwards curves remove the overhead of checking for exceptional cases, and twisted Edwards form removes the overhead of checking for invalid inputs. In this paper, we study the possibility of using isogenies of Edwards curves in the SIDH protocol, and study its potential speed and security benefits. Our results indicate that although Montgomery curves are faster for SIDH computations, the completeness of Edwards curves formulae provides additional security benefits against side-channel attacks. Since SIDH is still in its infancy, it is unclear if exceptional cases could be used as the basis for a side-channel attack, but in any case our EdSIDH implementation defends against this possibility.

Our contributions can be summarized as follows:

- We propose EdSIDH: fast formulas for SIDH over Edwards curves.
- We investigate isogeny formulas on projective and completed Edwards forms.
- We propose fast formulas for Edwards curve isogenies of degree 2, 3, and 4.

The rest of the paper is organized as follows: In the rest of this section, we provide preliminaries of Edwards curves and review the SIDH protocol. In Section 2, we provide new formulae for a key exchange scheme based on Edwards

curves. In Section 3, we present fast equations for EdSIDH arithmetic and analyze their running time complexity in terms of operation counts. In Section 4, we analyze the complexity of incorporating our Edwards arithmetic in SIDH. Finally, we conclude the paper in Section 5.

Independent work on fast isogeny formulas for Edwards curves was done in [19].

1.1 The Edwards Form

In 2007, Edwards introduced a new model for elliptic curves [15] called Edwards curves. Twisted Edwards curves are a generalization of Edwards curves, with each twisted Edwards curve being a quadratic twist of an Edwards curve. Twisted Edwards curves are defined by the equation $E_{a,d} : ax^2 + y^2 = 1 + dx^2y^2$ over a field $\mathbb{K}$, with $d \neq 0, 1; a \neq 0$. When $a = 1$, the curve defined by $E_{a,d}$ is an Edwards curve.

The isomorphism $(x, y) \mapsto \left(\frac{x}{\sqrt{a}}, y\right)$ maps the twisted Edwards curve $E_{a,d}$ to the isomorphic Edwards curve $E_{1,d/a}$, with the inverse map given by $(x, y) \mapsto (\sqrt{a}x, y)$ [4]. Over finite fields, only curves with order divisible by 4 can be expressed in the (twisted) Edwards form.

The group addition law on twisted Edwards curves is defined by:

$$(x_1, y_1) + (x_2, y_2) = \left(\frac{x_1y_2 + x_2y_1}{1 + dx_1x_2y_1y_2}, \frac{y_1y_2 - ax_1x_2}{1 - dx_1x_2y_1y_2}\right), \tag{1}$$

with identity element $(0, 1)$. If $\frac{a}{d}$ is not a square in $\mathbb{K}$, then the twisted Edwards addition law is strongly unified and complete: it can be used for both addition and doubling, and has no exceptional points. Additionally, when this is the case, the curve $E_{a,d}$ has no singular points. These properties of Edwards curves have in the past proved valuable, and have been used for simpler implementations and protection against side channel attacks in various cryptographic protocols [6].

However, if $\frac{a}{d}$ is not a square, then $E_{a,d}$ has only one point of order 2, namely $(0, -1)$ [5, Theorem 3.1]. As we will see later, the SIDH protocol is based on the repeated computation of 2-isogenies (with the private key defined as a point of order 2^k). As such, a unique point of order 2 would compromise the scheme's security, which means we must consider curves where $\frac{a}{d}$ is a square in $\mathbb{K}$. In the next section, we consider the additional points that occur when $\frac{a}{d}$ is a square, and present curve embeddings that allow us to desingularize these points.

In the case where $\frac{a}{d}$ is not a square, it is often useful to consider the dual addition law for Edwards curves:

$$(x_1, y_1) + (x_2, y_2) \mapsto \left(\frac{x_1y_1 + x_2y_2}{y_1y_2 + ax_1x_2}, \frac{x_1y_1 - x_2y_2}{x_1y_2 - y_1x_2}\right). \tag{2}$$

The addition law and dual addition law return the same value if both are defined. Additionally, for any pair of points on a twisted Edwards curve, at least one of the two addition laws will be defined.

1.2 Projective Curves and Completed Twisted Edwards Curves

If $\frac{a}{d}$ is a square in $\mathbb{K}$, then there are points $(x_1, y_1), (x_2, y_2)$ on the curve $E_{a,d}$ for which $(1 - dx_1x_2y_1y_2)(1 + dx_1x_2y_1y_2) = 0$ and the group law is not defined. We can embed the curve into projective space, add new singular points at infinity and generalize the group law to work for the new embedding, as is often done. We consider two representations of points on twisted Edwards curves, namely projective coordinates and completed coordinates.

The projective twisted Edwards curve is defined by $aX^2Z^2 + Y^2Z^2 = Z^4 + dX^2Y^2$. The projective points are given by the affine points, embedded as usual into P^2 by $(x, y) \mapsto (x : y : 1)$, and two extra points at infinity, $(0 : 1 : 0)$ of order 4, and $(1 : 0 : 0)$ of order 2. A projective point $(X : Y : Z)$ corresponds to the affine point $(x, y) = (\frac{X}{Z}, \frac{Y}{Z})$. Adding a generic pair of points takes $10M + 1S + 1A + 1D$ operations, and doubling takes $3M + 4S + 1A$ operations [5].

The completed twisted Edwards curve is defined by the equation:

$$\bar{E}_{a,d} := aX^2T^2 + Y^2Z^2 = Z^2T^2 + dX^2Y^2 \tag{3}$$

The completed points are given by the affine points embedded into $\mathbb{P}^1 \times \mathbb{P}^1$ via $(x, y) \mapsto ((x : 1), (y : 1))$, and up to four extra points at infinity, $((1 : 0), (\pm\sqrt{\frac{a}{d}} : 1))$ and $((1 : \pm\sqrt{d}), (1 : 0))$ [7]. The affine equivalent of a completed point $((X : Z), (Y : T))$ is given by $(x, y) = (\frac{X}{Z}, \frac{Y}{T})$.

If $P_1 = ((X_1 : Z_1), (Y_1 : T_1))$ and $P_2 = ((X_2 : Z_2), (Y_2 : T_2))$, then the group law is defined as follows:

$$\begin{aligned}
X_3 &= X_1Y_2Z_2T_1 + X_2Y_1Z_1T_2 & X_3' &= X_1Y_1Z_2T_2 + X_2Y_2Z_1T_1,\\
Z_3 &= Z_1Z_2T_1T_2 + dX_1X_2Y_1Y_2 & Z_3' &= aX_1X_2T_1T_2 + Y_1Y_2Z_1Z_2,\\
Y_3 &= Y_1Y_2Z_1Z_2 - aX_1X_2T_1T_2 & Y_3' &= X_1Y_1Z_2T_2 - X_2Y_2Z_1T_1,\\
T_3 &= Z_1Z_2T_1T_2 - dX_1X_2Y_1Y_2 & T_3' &= X_1Y_2Z_2T_1 - X_2Y_1Z_1T_1.
\end{aligned}$$

Hence we have $X_3Z_3' = X_3'Z_3$ and $Y_3T_3' = Y_3'T_3$, with either $(X_3, Z_3) \neq (0, 0)$ and $(Y_3, T_3) \neq (0, 0)$ or $(X_3', Z_3') \neq (0, 0)$ and $(Y_3', T_3') \neq (0, 0)$. We set $P_1 + P_2 = P_3$, where P_3 is either $((X_3 : Z_3), (Y_3 : T_3))$ or $((X_3' : Z_3'), (Y_3' : T_3'))$, depending on which of the above equations holds. With the identity point $((0 : 1)(1 : 1))$, the above defines a complete set of addition laws for complete twisted Edwards curves. This result formalizes the combination of the affine and dual addition law into a single group law.

The following result from Bernstein and Lange in [7] allows us to categorize pairs of points for which each addition law is defined:

When computing the result of $P + Q$, the original addition law fails exactly when $P - Q = ((1 : \pm\sqrt{d}), (1 : 0))$ or $P - Q = ((1 : 0), (\pm\sqrt{a/d} : 1))$. By the categorization of points of low even order from [5], the original addition law fails when $P - Q$ is a point at infinity of order 2 or 4. In particular, the original addition law is always defined for point doubling, as $P - P = O$, which has order 1.

The dual addition law fails exactly when $P - Q = ((1 : \pm\sqrt{a}), (0 : 1))$ or $P - Q = ((0 : 1), (\pm 1 : 1))$. In particular, the dual addition law fails exactly when $P - Q$ is a point of order $1, 2$ or 4 and is not a point at infinity.

We can use this categorization results to minimize the number of times we need to use the addition law for completed Edwards curves by considering order of the pairs of points involved in each section of the EdSIDH protocol.

1.3 Isogenies and Isogeny Computation

Isogenies are defined as structure preserving maps between elliptic curves. They are given by rational maps between the two curves, but can be equivalently defined by their kernel. If this kernel is generated by a point of order ℓ, then the isogeny is known as an ℓ-isogeny. In [31], Vélu explicitely showed how to find the rational functions defining an isogeny for an elliptic curve in Weierstrass form, given the kernel F.

The computation of isogenies of large degree can be reduced to the computation of smaller isogenies composed together, as described in [14]. For instance, consider computing an isogeny of degree ℓ^e. We reduce it to e computations of degree ℓ isogenies by considering a point $R \in E$ of degree ℓ^e that generates the kernel. We start with $E_0 := E, R_0 := R$ and iteratively compute $E_{i+1} = E_i/\langle l^{e-i-1} R_i \rangle$, $\phi_i \colon E_i \to E_{i+1}$, $R_{i+1} = \phi_i(R_i)$, using Vélu's formulas to compute the ℓ-isogeny at each iteration.

1.4 A review of Isogeny-Based Key-Exchange

Fix two small prime numbers ℓ_A and ℓ_B and an integer cofactor f, and let p be a large prime of the form $p = \ell_A^{e_A} \ell_B^{e_B} f \pm 1$ for some integers e_A, e_B. Let E be a supersingular elliptic curve defined over $\mathbb{F}_{p^2}$ which has group order $(\ell_A^{e_A} \ell_B^{e_B} f)^2$. All known implementations to date choose $\ell_A = 2, \ell_B = 3$ and $f = 1$, although other choices of ℓ_A, ℓ_B are possible. Public parameters consist of the supersingular elliptic curve E, and bases $\{P_A, Q_A\}$ and $\{P_B, Q_B\}$ of $E[\ell_A^{e_A}]$ and $E[\ell_B^{e_B}]$ respectively. During one round of key-exchange, Alice chooses two secret, random elements $m_A, n_A \in \mathbb{Z}/\ell_A^{e_A}\mathbb{Z}$, not both divisible by ℓ_A, and computes an isogeny $\phi_A \colon E \to E_A$ with kernel $K_A := \langle [m_A]P_A, [n_A]Q_A \rangle$. She also computes the image $\phi_A(P_B), \phi_A(Q_B)$ of the basis $\{P_B, Q_B\}$. Similarly, Bob selects random elements $m_B, n_B \in \mathbb{Z}/l_B^{e_B}\mathbb{Z}$, and computes an isogeny $\phi_B \colon E \to E_B$ with kernel $K_B := \langle [m_B]P_B, [n_B]Q_B \rangle$, along with the points $\phi_B(P_A), \phi_B(Q_A)$. After receiving $E_B, \phi_B(P_A), \phi_B(Q_A)$, Alice computes an isogeny $\phi'_A \colon E_B \to E_{AB}$ with kernel $\langle [m_A]\phi_B(P_A), [n_A]\phi_B(Q_A) \rangle$. Bob proceeds similarly to obtain a curve E_{BA} that is isomorphic to E_{AB}. Alice and Bob then use as their shared secret the j-invariant common to E_{BA} and E_{AB}. For more about the key exchange based on isogenies, please refer to [14].

2 EdSIDH

In this section, we provide even and odd isogenies over Edwards curves and propose a new formulation for SIDH, which we call EdSIDH moving forward. Here, we use M, S, C to refer to the cost of a multiplication, squaring, and multiplication by a curve constant in $\mathbb{F}_{p^2}$. We will also use R to refer to the cost of a square root, and I to the cost of an inversion. As is usually done, we ignore the cost of addition and subtraction as the cost is significantly smaller than the cost of multiplication and inversion.

2.1 Odd Isogenies in Edwards Form

In [29], Moody and Shumow presented ℓ-isogeny formulas for odd ℓ on Edwards curves. Let the subgroup $F = \langle(\alpha, \beta)\rangle = \{(0,1), (\pm\alpha_1, \beta_1), \ldots, (\pm\alpha_s, \beta_s)\}$ be the kernel of the desired ℓ-isogeny, with $\ell = 2s + 1$ and (α, β) a point of order ℓ on the curve E_d that generates F. Then

$$\psi(P) = \left(\prod_{Q \in F} \frac{x_{P+Q}}{y_Q}, \prod_{Q \in F} \frac{y_{P+Q}}{y_Q} \right) \tag{4}$$

maps E_d to $E_{d'}$, where $d' = B^8 d^\ell$ and $B = \prod_{i=1}^{s} \beta_i$.

If d is not a square in $\mathbb{K}$, the affine addition law is defined everywhere. Note that any odd isogeny from a curve with d not square maps to a curve with d' not square, as for an odd ℓ $d' = B^8 d^\ell$ is a square if and only if d is a square. This implies that if we chain odd isogenies starting with the curve E_d with d not a square in $\mathbb{K}$, then the affine addition law will be defined for any pair of points on any Edwards curve in the chain as they will all have a non-square coefficient.

The next proposition shows that the affine addition law is defined for all pairs of points in an odd isogeny computation even if d is not a square in $\mathbb{K}$.

Proposition 1. *The affine addition law is defined for all point additions in the EdSIDH protocol.*

Proof. During the EdSIDH protocol, we need to evaluate each 3-isogeny three times: on the current kernel point of order 3^k for some $k \leq e_A$, and on Alice's public points P_A, Q_A of order 2^{ℓ_A}. When evaluating $\psi(P)$, we must compute $P + Q$ for all $Q \in F$ (note that all such Q's have odd order). These are the only additions we need to do in order to compute an ℓ-isogeny. We now consider a few cases that cover these additions.

If P and Q both have odd order, then $-Q$ also has odd order, and $P - Q$ must have odd order as the order divides $lcm(ord(P), ord(Q))$. Therefore, it cannot be equal to a point at infinity as they have order either 2 or 4. Thus, by the categorization of exceptional points for the group law in section 1.2, we can compute $P + Q$ using the affine addition law.

Similarly, if P has even oder 2^{ℓ_A}, we note that $gcd(ord(P), ord(-Q)) = 1$ for all Q in the kernel of the 3 isogeny. Hence, we have that $ord(P - Q) =$

$lcm(ord(P), ord(-Q)) = lcm(ord(P), ord(Q))$. As $ord(Q)$ is odd, this implies $P - Q$ is not a point of order 2 or 4 (if $ord(Q) = 1 \implies Q = O$, then the affine addition law is always defined for $P + Q$).

Thus, in all cases, we can use the original addition law to compute and evaluate a 3-isogeny.

We can use the affine addition law to derive explicit coordinate maps for a ℓ-isogeny with kernel F (where $\ell = 2s + 1$):

$$\psi(x, y) = \left(\frac{x}{B^2} \prod_{i=1}^{s} \frac{\beta_i^2 x^2 - \alpha_i^2 y^2}{1 - d^2 \alpha_i^2 \beta_i^2 x^2 y^2}, \frac{y}{B^2} \prod_{i=1}^{s} \frac{\beta_i^2 y^2 - \alpha_i^2 x^2}{1 - d^2 \alpha_i^2 \beta_i^2 x^2 y^2} \right)$$

Moody and Shumow also presented an ℓ-isogeny formula for twisted Edwards curves. However, since each twisted Edwards curve is isomorphic to an Edwards curve, and the even isogeny formulas presented later output Edwards curves (with $a = 1$), one can use the isogeny formulas for Edwards curves (which are slightly faster to compute).

2.2 Even Isogenies in Edwards Form

In [29], Moody and Shumow presented Edwards curves isogenies formulas for isogenies with kernel $\{(0, 1), (0, -1)\}$. We generalize their work in two ways. First, we extend their formulas to work for 2-isogenies on completed twisted Edwards curves with arbitrary kernels. Then we show how to calculate 4-isogenies on Edwards curves. Finally, we consider methods for decreasing the computation cost for even isogenies in EdSIDH.

Suppose we want to compute an isogeny with kernel $\langle P_2 \rangle$, where P_2 is a point of order 2 on $E_{a,d}$. We follow an approach similar to that given in [14]. Since we already know how to calculate 2-isogenies with kernel $\{(0, 1), (0, -1)\}$, we find an isomorphism that maps P_2 to $(0, -1)$ and then use one of Moody's [29] isogeny formulas.

Proposition 2. *There exists an isomorphism between complete twisted Edwards curves that maps a point P_2 of order 2 to the point $(0, -1)$.*

Proof. We construct the desired isomorphism as follows. An isomorphism between the complete Edwards curve $\bar{E}_{a,d}$ and the Montgomery curve $E_{A,B} : By^2 = x^3 + Ax^2 + x$ (in projective coordinates) is given in [7] by:

$$\phi\colon ((X : Z), (Y : T)) \mapsto \begin{cases} (0 : 0 : 1) \text{ if } ((X : Z), (Y : T)) = ((0 : 1), (-1 : 1)) \\ ((T + Y)X : (T + Y)Z : (T - Y)X) \text{ otherwise} \end{cases}$$

$$\phi^{-1}\colon (U : V : W) \mapsto \begin{cases} ((0 : 1), (1 : 1)) & (U : V : W) = (0 : 1 : 0) \\ ((0 : 1), (-1 : 1)) & (U : V : W) = (0 : 0 : 1) \\ ((U : V), (U - W : U + W)) & \text{otherwise} \end{cases}$$

where $A = \frac{2(a+d)}{(a-d)}, B = \frac{4}{(a-d)}$ (and $a = \frac{A+2}{B}, d = \frac{A-2}{B}$). This isomorphism maps the point $(0,-1)$ to $(0,0)$, and vice versa.

An isomorphism between Montgomery curves mapping any point (x_2, y_2) of order 2 to $(0,0)$ and a point (x_4, y_4) doubling to it to the point $(1, \ldots)$ is presented in [14, Equation (15)]:

$$\phi_2 \colon (x,y) \mapsto \left(\frac{x - x_2}{x_4 - x_2}, \frac{y}{x_4 - x_2} \right) \tag{5}$$

The new curve has equation $E' : \frac{B}{x_4 - x_2} y^2 = x^3 + \frac{3x_2 + A}{x_4 - x_2} x^2 + x$.

Since ϕ, ϕ^{-1}, and ϕ_2 are isomorphisms, $\phi^{-1} \cdot \phi_2 \cdot \phi$ is also an isomorphism. Thus, we get an isomorphism mapping any point of order 2 to $((0:1)(-1:1))$ on $\bar{E}_{a,d}$.

The resulting curve has coefficients

$$a' = \frac{[x_2 + 2x_4](a-d) + 2(a+d)}{4} \qquad d' = \frac{[5x_2 - 2x_4](a-d) + 2(a+d)}{4},$$

where x_2 and x_4 are the x-coordinates of the point of order 2 and 4 on the Montgomery curve. These coordinates can be retained from the isogeny computation, and thus can be used here at no cost. The map $(x,y) \mapsto (\frac{x}{2}, y)$ maps the curve $E_{a',d'}$ to the curve $E_{4a',4d'}$, thus removing the inversion. The curve coefficients can thus be calculated in $2M$.

By using projective coordinates, we can calculate this isomorphism in $14M$ operations. Mapping a completed point to the Montgomery curve takes $3M$ operations, and $2M$ operations if we only need the X, Z coordinates (as is the case for the points of order 2 and 4), for a total of $7M$ to map all points to the Montgomery curves. The isomorphism ϕ_2 then takes $7M$ operations in projective coordinates, and the isomorphism back to an Edwards curve does not involve any addition. Thus, the total operations needed (ignoring addition and subtraction as is usually done) is $14M$ operations.

To calculate an arbitrary 2-isogeny of Edwards curves, we can first use the isomorphism presented above, and then apply one of the three Edwards curve 2-isogenies presented in [29].

2-isogenies on Edwards Curves. All 2-isogeny equations given by Moody and Shumow [29] require the computation of a square root, which makes them ill-suited to the SIDH framework, as many of them need to be calculated. However, when we know a point P_8 of order 8 such that $4P_8 = (0,-1)$, we can find a square root-free 2-isogeny formula for Edwards curves.

Consider a twisted Edwards curve $E_{a,d} : ax^2 + y^2 = 1 + dx^2y^2$. A birational transformation sending $E_{a,d}$ to the curve $E : y^2 = x^3 + 2(a+d)x^2 + (a-d)^2 x$ is given by:

$$\phi_1 \colon (x,y) \mapsto \left((a-d)\frac{1+y}{1-y}, (a-d)\frac{2(1+y)}{x(1-y)} \right)$$

By Vélu's formulas [31], a 2-isogeny on this curve with kernel $\{(0,0),\infty\}$ is given by:

$$\phi_2 \colon (x,y) \mapsto \left(\frac{x^2+(a-d)^2}{x}, y\frac{x^2-(a-d)^2}{x^2}\right)$$

The equation for the resulting curve is

$$E' : y^2 = x^3 + 2(a+d)x^2 - 4(a-d)^2x - 8(a+d)(a-d)^2$$

Using one of the points of order 2 on this curve, we can map it to a curve of the form $y^2 = x^3 + ax^2 + x$. For instance, the point $(2(a-d),0)$ has order 2, and the transformation $(x,y) \mapsto (x-2(a-d),0)$ maps the curve E' to the curve

$$E'' : y^2 = x^3 - 4(d-2a)x^2 + 16(a-d)x$$

Now, if we have a point of order 4 (r_1, s_1), the map $\phi_3 \colon (x,y) \mapsto \left(\frac{s_1 x}{r_1 y}, \frac{x-r_1}{x+r_1}\right)$ maps to the curve $x^2 + y^2 = 1 + d'x^2y^2$, where $d' = 1 - \frac{4r_1^3}{s_1^2}$.

If we evaluate the point P_8 of order 8 through the first three maps, we obtain a point of order 4 on the curve E'', since the 2-isogeny brings $4P_8 = (0,0)$ to the identity point. Doing so, we can obtain explicit equations for a 2-isogeny.

Consider a point $P_8 = (\alpha, \beta)$ of order 8 on the curve $E_{a,d}$ (Note that P_8 can be written in affine form, as all singular points have order 2 or 4). Then we have that

$$(\alpha_1, \beta_1) = \left((-4\frac{\beta^2(a-d)}{\beta^2-1}, 8\frac{\beta(a-d)}{\alpha(1-\beta^2)}\right)$$

is a point of order 4 on the curve $y^2 = x^3 - 4(d-2a)x^2 + 16(a-d)x$. We obtain $d' = 1 + 4\frac{\beta^4\alpha^2(a-d)}{\beta^2-1}$. Thus, a 2-isogeny mapping the curve $E_{a,d}$ to the curve $E_{1,d'}$ is given by:

$$(x,y) \mapsto \left(\frac{xy}{\alpha\beta}, \frac{x(\beta^2-1)+4\beta^2(a-d)}{x(\beta^2-1)-4\beta^2(a-d)}\right)$$

In the SIDH key-exchange calculations, a point of order 8 will be known for all but the last two isogeny calculations, as we are calculating an isogeny with kernel generated by a point of order 2^{e_A}, with e_A large.

Recall that in the SIDH protocol, Alice selects an element $R_A = [m_A]P_A + [n_A]Q_A$ of the elliptic curve E of order 2_A^e, which generates the kernel of the isogeny ϕ_A. She computes the isogeny iteratively, one 2 or 4 isogeny at a time. Consider one step in this process:

Suppose R'_A is a point of order 2^{e_A-k} on the curve E', a k-isogeny of the original curve E. For the next step in the iteration, Alice computes the points $R''_A = 2^{e_A-k-3}R_A$ and $4R''_A = 2^{e_A-k-1}R_A$. We have that $4R''_A$ is a point of order 2 on the curve E', with R''_A a point of order 8 above it. Thus, we can use these points to calculate a 2-isogeny with kernel $4R''_A$, as described above.

4-isogenies on Edwards Curves. Let us assume we are given a twisted Edwards curve $E_{a,d}$ and a point P_4 on the curve of order 4. We want to calculate a 4-isogeny on the curve with kernel generated by P_4, without knowing a point of order 8 that doubles to P_4. We can do so as follows: Use the isomorphism presented earlier to map P_4 and $2P_4$ to $((1:\sqrt{a'}),(0:1)),((0:1),(-1:1))$ respectively, on some isomorphic curve $E_{a',d'}$. Then use the isomorphism $(x,y) \mapsto (\frac{x}{\sqrt{a'}}, y)$ to map the curve to $E_{1,\frac{d'}{a'}}$. Finally, compose the following two 2-isogeny formulas of Moody and Shumow [29] to calculate the 4-isogeny:

$$\phi_1(x,y) \mapsto \left((\gamma \pm 1)xy, \frac{(\gamma \mp 1)y^2 \pm 1}{(\gamma \pm 1)y^2 \mp 1}\right)$$
$$\phi_2(x,y) \mapsto \left(i(\rho \mp 1)\frac{x}{y}\frac{1-d'y^2}{1-d'}, \frac{d' \mp \rho\,\rho y^2 \pm 1}{d' \pm \rho\,\rho y^2 \mp 1}\right)$$

that map $E_{1,d}$ to $E_{1,d'}$ with $d' = (\frac{\gamma\pm1}{\gamma\mp1})^2$ and $E_{1,d'}$ to $E_{1,\hat{d}}$ with $\hat{d} = (\frac{\rho\pm1}{\rho\mp1})^2$, where $\gamma^2 = 1-d$ and $\rho^2 = d', i^2 = -1$ in $\mathbb{K}$. Note that d' is, by definition, a square in $\mathbb{K}$ and so the curve $E_{1,d'}$ will have singular points and exceptions to the group law.

Both isogenies have kernel $\{((0:1),(-1:1)),((0:1),(1:1))\}$ and the first isogeny maps $((1:\sqrt{a'}),(0:1))$ to $((0:1),(-1:1))$, so the composition is well defined as a 4-isogeny with kernel generated by $((1:\sqrt{a'}),(0:1))$. Composing the two equations for the curve coefficient, we get:

$$\hat{d} = \left(\frac{\rho \pm 1}{\rho \mp 1}\right)^2 = \left(\frac{(\frac{\gamma\pm1}{\gamma\mp1}) \pm 1}{(\frac{\gamma\pm1}{\gamma\mp1}) \mp 1}\right)^2 = \left(\frac{(\gamma \pm 1) \pm (\gamma \mp 1)}{(\gamma \pm 1) \mp (\gamma \mp 1)}\right)^2$$

which costs one square root and one inversion. The value of $i = \sqrt{-1}$ in $\mathbb{K}$ can be computed and stored ahead of time to evaluate 4-isogenies.

3 EdSIDH Arithmetic

Here we describe our explicit formulas for fast isogenies of degree 2,3, and 4 for Edwards curves.

3.1 Point multiplication by ℓ

Let P be a point on our curve and ℓ an integer, and suppose we want to compute ℓP. By [5], we know that the affine group law is always defined for point doublings (even when d is a square in the field K). To compute this, we can use a ladder algorithm, which takes n steps (where n is the number of bits of ℓ), each consisting of a doubling and a point addition.

On a projective curve, we know from [6] that we can double a point by $3M + 4S$, and adding arbitrary points takes $10M + 1S + 1C$. On complete curves, doubling takes $5M + 4S + 1C$, and addition takes $29M$ operations.

3.2 Computing 3-isogenies

In the case where $a = 1$ and d is not a square in K, Moody and Shumow [29] presented a way to calculate a 3-isogeny in projective form with kernel $\{(0,0),(\pm A, B, 1)\}$ at a cost of $6M + 4S + 3C$. Generalizing to the case where $P_3 = (\alpha, \beta, \zeta)$ is a point of order 3 (with $A = \alpha/\zeta, B = \beta/\zeta$), and we want to evaluate the 3-isogeny with kernel $\langle P_3 \rangle$ on a generic projective point (x, y, z), we get the following equations for the evaluation of the 3-isogeny:

$$\psi(x,y,z) = (xz\gamma^4(\beta^2x^2 - \alpha^2y^2), yz\gamma^4(\beta^2y^2 - \alpha^2x^2), \beta^2(\gamma^4z^4 - d^2x^2y^2\alpha^2\beta^2)))$$

It takes $13M + 9S$ operations to compute $\psi(x, y, z)$. If we are evaluating the isogeny at multiple points, we don't need to recompute $\alpha^2, \beta^2, \gamma^2, \gamma^4, d^2$, thus bringing the cost to $13M + 4S$ for each additional point evaluation.

We can compute the curve coefficient $d' = \beta^8 d^3$ by computing $\beta^8 = ((\beta^2)^2)^2$ and $d^3 = d^2 d$ for a total cost of $3S + 2M$, or $4S + 2M$ if we didn't evaluate the isogeny ahead of time.

3.3 Computing 2-isogenies

Let us consider the 2-isogeny equation presented in section 2.2, where (α, β) is a point of order 8 on the curve $E_{a,d}$.

$$(x, y) \mapsto \left(\frac{xy}{\alpha\beta}, \frac{x(\beta^2 - 1) + 4\beta^2(a - d)}{x(\beta^2 - 1) - 4\beta^2(a - d)}\right)$$

We can compute it using $2I + 7M + 1S$ or $I + 10M + 1S$ with a simultaneous inversion. Alternatively, we can define an equivalent version for completed coordinates by representing $x = \frac{X}{Z}, y = \frac{Y}{T}, \alpha = \frac{A}{Z_P}, \beta = \frac{B}{T_P}$:

$$\begin{aligned}((X : Y), (Z : T)) \mapsto ((XYZ_PT_P : ABZT),\\ (X(B^2 - T_P^2) + 4B^2(a - d)Z : X(B^2 - T_P^2) - 4B^2(a - d)Z))\end{aligned}$$

Precomputing shared subexpressions allows us to compute this in $9M + 2S$ operations. Combined with the $14M$ operations for the isomorphism bringing any point of order 2 to $(0, -1)$, we get a total of $23M + 2S$ operations.

We could also compute this isogeny using projective coordinates, where $x = \frac{X}{Z}, y = Y, Z, \alpha = \frac{A}{Z_0}, \beta = \frac{B}{Z_0}$:

$$\begin{aligned}(X : Y : Z) \mapsto (XYZ_0^2, X(B^2 - Z_0^2) + 4B^2Z(a - d),\\ Z^2A^2B^2(X(B^2 - Z_0^2) - 4B^2Z(a - d)))\end{aligned}$$

which can be computed in $7M + 3S$ operations. Combining this with the $14M$ operations for the isomorphism bringing any point of order 2 to $((0 : 1), (-1 : 1))$ and the map $((X : Z), (Y : T)) \mapsto (XT, YZ, TZ)$ $(3M)$ that embeds a completed

point into a projective curve, we get a total cost of $24M + 3S$ (which is more expensive than using completed coordinates).

The curve coefficient is given by $d' = 1 + 4\frac{\beta^4\alpha^2(a-d)}{\beta^2-1}$. This can be computed in $5M + 1I$ operations. Combining this with the $2M$ operations used to compute the curve coefficients from the isomorphism, we get a total of $7M + 1I$ operations.

3.4 Computing 4-isogenies

Recall the 4-isogeny formulas presented in the section 2.2

$$\phi_1(x, y) \mapsto \left((\gamma \pm 1)xy, \frac{(\gamma \mp 1)y^2 \pm 1}{(\gamma \pm 1)y^2 \mp 1}\right)$$

that maps $E_{1,d}$ to $E_{1,d'}$ with $d' = (\frac{\gamma\pm1}{\gamma\mp1})^2$ where $\gamma^2 = 1 - d$, and

$$\phi_2(x, y) \mapsto \left(i(\rho \mp 1)\frac{x}{y}\frac{1 - d'y^2}{1 - d'}, \frac{d' \mp \rho}{d' \pm \rho}\frac{\rho y^2 \pm 1}{\rho y^2 \mp 1}\right)$$

that maps $E_{1,d'}$ to $E_{1,\hat{d}}$ with $\hat{d} = (\frac{\rho\pm1}{\rho\mp1})^2$, where $\rho^2 = d', i^2 = -1$ in $\mathbb{K}$. We can rewrite these in $\mathbb{P}_1 \times \mathbb{P}_1$, writing $x = \frac{X}{Z}, y = \frac{Y}{T}$ as follows:

$$\phi_1((X, Z), (Y, T)) \mapsto (((\gamma \pm 1)XY, ZT), ((\gamma \mp 1)Y^2 \pm T^2, (\gamma \pm 1)Y^2 \mp T^2))$$

and

$$\phi_2((X, Z), (Y, T)) \mapsto (((i(\rho \mp 1)XT(T^2 - dY^2), YZT^2(1 - d)), \\ ((d \mp \rho)(\rho Y^2 \pm T^2), (d \pm \rho)(\rho Y^2 \mp T^2)))$$

We can compute ϕ_1 in $7M$ operations, and ϕ_2 in $13M$ operations. Adding the cost of the isomorphism that brings our point of order 4 to $((1 : \sqrt{a'}), (0 : 1))$, we get a total cost of $34M$ to evaluate a 4-isogeny.

Due to the complete lack of symmetry between the x and y coordinates in both the ϕ_1 and ϕ_2 maps, using projective coordinates takes even more operations than using completed coordinates (for instance evaluating ϕ_1 in projective coordinates would take $7M + 2S$ to compute). Hence, the fastest way to evaluate a 4-isogeny with points on projective coordinates would be to embed them in the complete curve (no cost), evaluate the isogeny, and map them back to a projective curve via the map $((X : Z), (Y : T)) \mapsto (XT, YZ, TZ)$ which takes $3M$ operations. The total cost is thus $37M$ operations.

Calculating the curve coefficient, given by $(\frac{(\gamma\pm1)\pm(\gamma\mp1)}{(\gamma\pm1)\mp(\gamma\mp1)})^2$ with $\gamma = \sqrt{1-d}$ additionally requires $1R + 1I + 1S$.

Since computing 4-isogenies is significantly more expensive than computing 2-isogenies due to the need to compute a square root, we propose using 2-isogenies whenever a suitable point of order 8 is known. In practice, this means we will only compute one 4-isogeny, at the very last iteration of isogeny computations.

Table 1. SIDH secret kernel generation cost per bit

Scheme	Cost per bit
Kummer Montgomery [14]	$9M + 6S$
Kummer Montgomery [16]	$6M + 4S$
Edwards with Montgomery Ladder	
Projective Edwards	$13M + 5S + 1C$
Complete Edwards	$34M + 4S + 1C$
Edwards with Window Method ($k = 4$)	
Projective Edwards	$5.5M + 4.25S + 0.25C$
Complete Edwards	$12.25M + 4S + 1C$

4 EdSIDH Computation Cost

Here, we analyze the full complexity to use Edwards curves for SIDH. Notably, we look at the cost of the large-degree isogeny computations, based on the operation costs presented in section 3.

4.1 Secret Kernel Generation

In SIDH, the secret kernel is generated from the double-point multiplication $R = nP + mQ$. However, as noted by [14], we can choose any such generator formula, including $R = P + mQ$. This formulation for a double-point multiplication greatly reduces the total cost of the double-point multiplication. In particular, [14] describes a 3-point Montgomery differential ladder that can be used with Montgomery coordinates, at the cost of two differential point additions and one point doubling per step. Faz-Hernández et al. [16] recently proposed a right-to-left variant of the 3-point ladder that only requires a single differential point addition and a single point doubling per step.

For EdSIDH, a 3-point ladder is not necessary to perform $R = P + mQ$. We can first perform the mQ computation and then simply finish with a point addition with P. Two options to compute the mQ computation are the standard Montgomery powering ladder [28] or the window approach [6]. The Montgomery ladder is a constant set of an addition and doubling for each step, whereas the window approach with a k-bit window performs k point doublings and then an addition. In Table 1, we compare the relative costs per bit in the secret key for this double-point multiplication. Note that this cost per bit does not include the final point addition for $P + mQ$ as this operation is a constant cost.

Thus, as we can see, there is a slight speed advantage with using projective Edwards curves with the Window method. We note that there are some security implications when using the window method instead of the Montgomery ladder, which we do not discuss here.

4.2 Secret Isogeny Computation

The second part of the SIDH protocol involves a secret isogeny walk based on the secret kernel. In this computation we chain isogenies of degree ℓ with kernel points $\ell^{e-i-1}R_i$. To efficiently calculate these kernel representations, we used the combinatorics strategy from [14]. By using pivot points to traverse a one-way acyclic graph, we can create an optimal strategy that represents the least cost to compute the large-degree isogeny.

To evaluate our EdSIDH formulas against the known Montgomery formulas, we computed the costs of our point multiplication by ℓ and isogeny evaluation by ℓ. Based on the relative cost, we computed an optimal strategy based on the algorithm from [14]. We used this to calculate the total cost of a large-degree isogeny for our Edwards isogeny formulas as well as the Montgomery formulas from previous works. Table 2 compares the cost of various isogeny and elliptic curve operations and Table 3 represents the full and normalized cost of a large-degree isogeny for the primes listed. We chose the primes $p_{503} = 2^{250}3^{159} - 1$ and $p_{751} = 2^{372}3^{239} - 1$ which have a quantum security of 83 and 124 bits, respectively.

As these tables show, Edwards arithmetic is a fair bit slower than Montgomery arithmetic. Large-degree isogenies with base degree 2 or 3 appear to be 2-3 times slower and base degree 4 isogenies are about 10 times slower when comparing Edwards to Montgomery. Interestingly, isogenies of degree 3 appear to be more efficient than isogenies of degree 2 for Edwards curves.

Table 2. Affine isogeny formulas vs. projective isogenies formulas. For the first column, the isogeny computations follow the form: $2P$ for point doubling, 2coef for finding a new isogenous curve of degree 2, and 2pt for pushing an point through an isogeny of degree 2. For this work's columns, the first column is for projective Edwards coordinates and the second column is for completed Edwards coordinates.

Iso. Comp.	Affine Mont. [14]	Proj. Mont. [12]	Affine Ed. (This Work)	
			Proj.	Complete
$2P$	$3M + 2S$	-	$3M + 4S$	$5M + 4S + C$
2coef	$I + 4M + S + C$	-	$I + 7M$	$I + 7M$
2pt	$2M + 1S$	-	$24M + 3S$	$23M + 2S$
$3P$	$7M + 4S$	$7M + 5S$	$13M + 5S + C$	-
3coef	$I + 4M + S + C$	$2M + 3S$	$2M + 4S$	-
3pt	$4M + 2S$	$4M + 2S$	$13M + 9S$	-
$4P$	$6M + 4S$	$8M + 4S$	$6M + 8S$	$10M + 8S + 2C$
4coef	$I + 2M + C$	$4S$	$R + I + S$	$R + I + S$
4pt	$6M + S$	$6M + 2S$	-	$34M$

Table 3. Normalized complexities for a large-degree isogeny computation for different coordinate schemes. We found the total cost of a large-degree isogeny for the formulas in Table 2 over isogenies with base 2,3, and 4. We then converted these costs from quadratic extension field arithmetic to the number of multiplications in the base prime field for easy comparison. Notably, we assumed that SIDH arithmetic is in $\mathbb{F}_{p^2}$ with irreducible modulus x^2+1 (as is the case in known implementations) for efficient computations.These are the total number of $\mathbb{F}_p$ multiplications ($\tilde{M}$), where $\mathbb{F}_{p^2}$operations are converted as follows: $R = 22\lceil \log_2 p \rceil \tilde{M}$, $I = 10\tilde{M}$, $M = 3\tilde{M}$, $S = 2\tilde{M}$, and $C = 2\tilde{M}$. We assumed an inversion was performed with extended Euclidean algorithm and the square root required two large exponentiations.

Large-Degree Isogeny	Affine Mont. [14]	Proj. Mont. [12]	Affine Ed. (This Work)	
			Proj.	Complete
2^{250}	$27102\tilde{M}$	-	$87685\tilde{M}$	$97841\tilde{M}$
3^{159}	$29686\tilde{M}$	$28452\tilde{M}$	$65355\tilde{M}$	-
4^{125}	$22617\tilde{M}$	$24126\tilde{M}$	$181582\tilde{M}$	$191278\tilde{M}$
2^{372}	$42516\tilde{M}$	-	$140454\tilde{M}$	$155450\tilde{M}$
3^{239}	$47650\tilde{M}$	$45864\tilde{M}$	$105469\tilde{M}$	-
4^{186}	$36118\tilde{M}$	$38842\tilde{M}$	$385756\tilde{M}$	$384732\tilde{M}$

5 Conclusions and Future Work

In this paper, we investigated employing Edwards curve for the supersingular isogeny Diffie-Hellman key exchange protocol and provided the required arithmetic and complexity analyses. Edward curves are attractive in the sense that they provide extra security benefits by having complete and unified addition formulae, which are not offered by Weierstrass and Montgomery forms.

Furthermore, we have seen that there are simple and elegant odd isogenies for Edwards curves. We note that an EdSIDH protocol with two odd primes would preserve a non-square curve coefficient and the completeness of the (simple) curve E_d for every isogeny computation. Because of this and the simple and fast formulas for odd isogenies presented, we suggest that Edwards curves would be a good choice for an odd-primes only implementation of SIDH.

Moving forward, we encourage cryptographic implementers to further investigate the performance of EdSIDH proposed in this paper for a fair and proper comparison to their counterparts. Integration of these formulas into SIKE [18] and static-static SIDH-like schemes [3] could also be interesting. Lastly, we will be following advances in side-channel attacks on isogeny-based schemes, such as those proposed in [21, 20], to see if our scheme will provide additional defense against such methods.

6 Acknowledgement

The authors would like to thank the reviewers for their comments. This work is supported in parts by awards NIST 60NANB16D246, NIST 60NANB17D184,

and NSF CNS-1801341. Also, this research was undertaken thanks in part to funding from the Canada First Research Excellence Fund, Natural Sciences and Engineering Research Council of Canada, CryptoWorks21, Public Works and Government Services Canada, and the Royal Bank of Canada.

References

1. Reza Azarderakhsh, Dieter Fishbein, and David Jao. Efficient implementations of a quantum-resistant key-exchange protocol on embedded systems. Technical report, 2014.
2. Reza Azarderakhsh, David Jao, Kassem Kalach, Brian Koziel, and Christopher Leonardi. Key compression for isogeny-based cryptosystems. In *Proceedings of the 3rd ACM International Workshop on ASIA Public-Key Cryptography*, AsiaPKC '16, pages 1–10, New York, NY, USA, 2016. ACM.
3. Reza Azarderakhsh, David Jao, and Christopher Leonardi. Post-quantum static-static key agreement using multiple protocol instances. In *Selected Areas in Cryptography – SAC 2017, 24th International Conference, Ottawa, ON, Canada, August 16-18, 2017, Revised Selected Papers*, pages 45–63, 2018.
4. Daniel J. Bernstein, Peter Birkner, Marc Joye, Tanja Lange, and Christiane Peters. Twisted Edwards curves. In Serge Vaudenay, editor, *Progress in Cryptology – AFRICACRYPT 2008*, pages 389–405, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.
5. Daniel J. Bernstein, Peter Birkner, Tanja Lange, and Christiane Peters. ECM using Edwards curves. *Math. Comp.*, 82(282):1139–1179, 2013.
6. Daniel J. Bernstein and Tanja Lange. Faster addition and doubling on elliptic curves. In *Advances in Cryptology - ASIACRYPT 2007, 13th International Conference on the Theory and Application of Cryptology and Information Security, Kuching, Malaysia, December 2-6, 2007, Proceedings*, pages 29–50, 2007.
7. Daniel J. Bernstein and Tanja Lange. A complete set of addition laws for incomplete edwards curves. *Journal of Number Theory*, 131(5):858 – 872, 2011. Elliptic Curve Cryptography.
8. Denis Charles, Kristin Lauter, and Eyal Goren. Cryptographic hash functions from expander graphs. *Journal of Cryptology*, 22(1):93–113, Jan 2009.
9. Lily Chen, Stephen Jordan, Yi-Kai Liu, Dustin Moody, Rene Peralta, Ray Perlner, and Daniel Smith-Tone. Report on post-quantum cryptography. Technical report, National Institute of Standards and Technology (NIST), April 2016.
10. Anamaria Costache, Brooke Feigon, Kristin Lauter, Maike Massierer, and Anna Puskas. Ramanujan graphs in cryptography. Cryptology ePrint Archive, Report 2018/593, 2018.
11. Costello Costello, Patrick Longa, and Michael Naehrig. Efficient algorithms for supersingular isogeny Diffie-Hellman. In *Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part I*, volume 9814 of *Lecture Notes in Computer Science*, pages 572–601, 2016.
12. Craig Costello and Huseyin Hisil. A simple and compact algorithm for SIDH with arbitrary degree isogenies. In *Advances in Cryptology - ASIACRYPT 2017 - 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part II*, pages 303–329, 2017.

13. Craig Costello, David Jao, Patrick Longa, Michael Naehrig, Joost Renes, and David Urbanik. Efficient compression of SIDH public keys. In *Advances in Cryptology - EUROCRYPT 2017 - 36th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Paris, France, April 30 - May 4, 2017, Proceedings, Part I*, pages 679–706, 2017.
14. Luca De Feo, David Jao, and Jérôme Plût. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. *Journal of Mathematical Cryptology*, 8(3):209–247, Sep. 2014.
15. Harold M. Edwards. A normal form for elliptic curves. In *Bulletin of the American Mathematical Society*, pages 393–422, 2007.
16. Armando Faz-Hernández, Julio López, Eduardo Ochoa-Jiménez, and Francisco Rodríguez-Henríquez. A faster software implementation of the Supersingular Isogeny Diffie-Hellman Key Exchange Protocol. *IEEE Transactions on Computers*, 2018. To appear.
17. Amir Jalali, Reza Azarderakhsh, Mehran Mozaffari-Kermani, and David Jao. Supersingular isogeny Diffie-Hellman key exchange on 64-bit ARM. *IEEE Trans. Dependable and Secure Computing I: Regular Papers*, 2017.
18. David Jao, Reza Azarderakhsh, Matthew Campagna, Craig Costello, Luca De Feo, Basil Hess, Amir Jalali, Brian Koziel, Brian LaMacchia, Patrick Longa, Michael Naehrig, Joost Renes, Vladimir Soukharev, and David Urbanik. Supersingular isogeny key encapsulation. Submission to the NIST Post-Quantum Standardization Project, 2017.
19. Suhri Kim, Kisoon Yoon, Jihoon Kwon, Seokhie Hong, and Young-Ho Park. Efficient isogeny computations on twisted Edwards curves. *Security and Communication Networks*, 2018.
20. Brian Koziel, Reza Azarderakhsh, and David Jao. An exposure model for supersingular isogeny Diffie-Hellman key exchange. In *Topics in Cryptology - CT-RSA 2018 - The Cryptographers' Track at the RSA Conference 2018, San Francisco, CA, USA, April 16-20, 2018, Proceedings*, pages 452–469, 2018.
21. Brian Koziel, Reza Azarderakhsh, and David Jao. Side-channel attacks on quantum-resistant supersingular isogeny Diffie-Hellman. In *Selected Areas in Cryptography – SAC 2017, 24th International Conference, Ottawa, ON, Canada, August 16-18, 2017, Revised Selected Papers*, pages 64–81, 2018.
22. Brian Koziel, Reza Azarderakhsh, David Jao, and Mehran Mozaffari-Kermani. On fast calculation of addition chains for isogeny-based cryptography. In *12th International Conference on Information Security and Cryptology, Inscrypt 2016*, volume 10143 of *Lecture Notes in Computer Science*, pages 323–342. Springer, 2016.
23. Brian Koziel, Reza Azarderakhsh, and Mehran Mozaffari-Kermani. Fast hardware architectures for supersingular isogeny Diffie-Hellman key exchange on FPGA. In *Progress in Cryptology – INDOCRYPT 2016: 17th International Conference on Cryptology in India, Kolkata, India, December 11-14, 2016, Proceedings*, pages 191–206, Cham, 2016. Springer International Publishing.
24. Brian Koziel, Reza Azarderakhsh, and Mehran Mozaffari-Kermani. A high-performance and scalable hardware architecture for isogeny-based cryptography. *IEEE Transactions on Computers*, PP(99):1–1, 2018.
25. Brian Koziel, Reza Azarderakhsh, Mehran Mozaffari-Kermani, and David Jao. Post-quantum cryptography on FPGA based on isogenies on elliptic curves. *IEEE Trans. Circuits and Systems I: Regular Papers*, 2016.

26. Brian Koziel, Amir Jalali, R. Azarderakhsh, David Jao, and Mehran Mozaffari-Kermani. NEON-SIDH: efficient implementation of supersingular isogeny Diffie-Hellman key exchange protocol on ARM. In *International Conference on Cryptology and Network Security*, pages 88–103. Springer, 2016.
27. Michael Meyer, Steffen Reith, and Fabio Campos. On hybrid SIDH schemes using Edwards and Montgomery curve arithmetic. Cryptology ePrint Archive, Report 2017/1213, 2017.
28. Peter L. Montgomery. Speeding the Pollard and elliptic curve methods of factorization. *Mathematics of Computation*, pages 243–264, 1987.
29. Dustin Moody and Daniel Shumow. Analogues of Vélu's formulas for isogenies on alternate models of elliptic curves. *Math. Comp.*, 85(300):1929–1951, 2016.
30. Vladimir Valyukh. Performance and comparison of post-quantum cryptographic algorithms. Master's thesis, Linkoping University, 2017.
31. Jacques Vélu. Isogénies entre courbes elliptiques. *C. R. Acad. Sci. Paris Sér. A-B*, 273:A238 – A241, 1971.
32. Youngho Yoo, Reza Azarderakhsh, Amir Jalali, David Jao, and Vladimir Soukharev. A post-quantum digital signature scheme based on supersingular isogenies. In Aggelos Kiayias, editor, *Financial Cryptography and Data Security*, pages 163–181, Cham, 2017. Springer International Publishing.